AD-A106 707   UNIVERSITY OF SOUTH FLORIDA TAMPA DEPT OF ELECTRICAL--ETC F/G 9/2
MICROPROCESSOR DIGITAL FAULT DIAGNOSIS.(U)
SEP 81   H A NIENHAUS   AFOSR-81-0163

UNCLASSIFIED   AFOSR-TR-81-0712   NL

END
DATE
FILMED
12 81
DTIC

AFOSR-TR- 81-0712

18 19

LEVEL II

C

6 MICROPROCESSOR DIGITAL FAULT DIAGNOSIS.

AD A106 707

10
H. A. Nienhaus
Electrical Engineering Department
Univeristy of South Florida
Tampa, Florida 33620

11 16 Sep [redacted] 81

12 47

9 Final Report, [redacted] 13 Mar [redacted] — 12 Sep [redacted] 81,

15 AFOSR-81-0163 16 2305 17 Ø9

Prepared for
Air Force Office of Scientific Research
Bldg. 410
Bolling AFB, D.C. 20332

81 11 06 055

408 755

Approved for public release;
distribution unlimited.

Unclassified

SECURITY CLASSIFICAT[illegible]HIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
AFOSR-TR- 81 -0712

2. GOVT ACCESSION NO.
AD-A106707

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Microprocessor Digital Fault Diagnosis

5. TYPE OF REPORT & PERIOD COVERED
Final
03/13/81 thru 09/12/81

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
H. A. Nienhaus

8. CONTRACT OR GRANT NUMBER(s)
AFOSR-81-0163

9. PERFORMING ORGANIZATION NAME AND ADDRESS
University of South Florida
Electrical Engineering Department
Tampa, Florida 33620

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
61102F
2305/D9

11. CONTROLLING OFFICE NAME AND ADDRESS
AFOSR, Bldg. 410
Bolling AFB, D.C. 20332

12. REPORT DATE
09/13/81

13. NUMBER OF PAGES
45

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release;
distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Digital Fault Diagnosis
Text Generator
Character Generator
Signature Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report gives a complete set of fault diagnosis flow diagrams for detecting any fault and isolating any single chip fault in a text generator, which contains a large variety of LS1, MS1, and SS1 ports. The fault diagnosis procedures are designed to minimize test processor diagnostic time and internal test points.

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

1.0. INTRODUCTION

This report documents the results of a study of the problems associated with a microprocessor-based automatic fault diagnosis system for a digital avionics subsystem consisting of a large variety of LS1, MS1, and SS1 parts. This work was performed under U.S. Air Force Grant No. AFOSR-81-0163. A text generator which is a subsystem of the Modular Programmable Display Generator (MPDG) for DAIS was selected as the test vehicle for this study.

The major objectives of the project were to establish a test philosophy and to isolate and develop techniques to solve as many fault diagnosis probelms as possible. These objectives have been met. Specifically, a unique test philosophy, which will be called system functional fault diagnosis, and a complete set of fault diagnosis flow diagrams to implement this test philosophy have been developed for the text generator. These are also applicable to other digital subsystems.

The fault diagnosis flow diagrams are capable of detecting any fault in the system, and isolating any single chip fault to within a maximum of 6 of 51 chips. In most cases, isolation to within 1 or 2 chips is possible. In addition, limited isolation of certain multiple faults is also possible. However, isolation of all multiple chip faults requires further study. Note that there are no restrictions on the type of single-chip fault that can be detected and isolated. This includes transient faults and multiple faults in a single function chip.

The major problems associated with existing automatic test equipment (ATE) are excessive processor diagnostic time and excessive interfaces between the test processor and the system under test. Consequently, the fault diagnosis procedures were designed primarily to minimize these. Several techniques were utilized to help accomplish this. Some of the more important are:

1) Initial fault detection and isolation is performed using the system primary inputs and outputs only. This enables rapid detection and partial isolation of faults since a large number of chips are exercised simultaneously.
2) Where practical, both initial and subsequent fault isolation is accomplished by breaking up the data into blocks, rather than breaking up the circuit into smaller blocks. In many cases, it is possible to isolate a fault to 1 or 2 chips without moving off the system bus.

3) Sequential fault diagnosis was utilized, i.e., the results of one test dictate what additional tests must be performed.

4) Signature analysis data compression techniques are used to compress large serial data strings consisting of thousands of bits into a 16 bit signature. This greatly reduces processor diagnostic time. Again the data block concept minimizes the need to move from test point to test point.

It is difficult to estimate the minimum and maximum diagnostic times required by the fault diagnosis flow diagrams without working out the details of the system. However, based upon past experience these should range from seconds to minutes. It is significant that the fault diagnosis procedures utilize only 10 internal and 19 external observation test points. No internal test points are used for control purposes.

In developing the fault diagnosis procedures, it soon became apparent that, in many cases, a greater degree of fault isolation can be obtained, without additional tests or test points, simply by adopting simple design for test rules. Consequently, these rules are also presented in this report. Unlike existing design for test rules, these rules also tend to minimize chips and system power requirements and increase system reliability.

2.0. TEXT GENERATOR DESCRIPTION AND OPERATION

Effective system functional fault diagnosis requires a thorough knowledge of how the system under test operates. For this reason, a simplified description of the text generator and its operation is included here. This description also makes it easier to follow the fault diagnosis flow diagrams presented in later sections. The text generator employs standard horizontal scan character generation techniques, and some knowledge of such techniques are assumed in this discussion.

A simplified block diagram of the text generator is shown in Fig. 2.1. The text generator module provides alpha-numeric text generation for two 525 line TV raster displays. One frame of text consists of 16 rows of 32 characters each. Sixty-four different ASC II characters with a 5 x 7 font are stored in ROMS A and B. The high addresses (A4-A9) of ROMS A and B determine the character that is to be displayed. These are addressed by 6 bit ASC II codes stored in RAMS A and B. For example, hex 17 which is the ASC II code for the letter O will address the character O in ROM. The low addresses (A0-A3) of ROMS A and B determine which line of the character addressed by RAM is to be displayed. These are addressed by the mod 12 line counter which provides 7 lines of character data followed by 5 blank lines for each character row. The 5 blank lines provide spacing between character rows.

The mod-6 dot counter loads character data from ROMS A and B into shift registers A and B, respectively. It then shifts out a serial stream consisting of a blank dot (for spacing between characters) followed by 5 horizontal character dots. It then sequences the mod-32 character counter to address the next character to be displayed in the horizontal line and the sequence is repeated. The character counter points to addresses A0-A3 in RAMS A and B. Thus one line of the entire 32 character row is displayed before the next line is addressed. After all 12 lines of a character row have been displayed, the line counter sequences the mod-16 row counter to address the next row to be displayed. The row counter points to addresses A4-A8 in RAMS A and B. This sequencing is repeated until an entire frame of 16 rows of 32 characters has been displayed.

An external dot clock is required to sequence the dot counter and shift registers. Also an external horizontal synchronizing pulse is required to start each new data line. This sequences the line counter and resets the

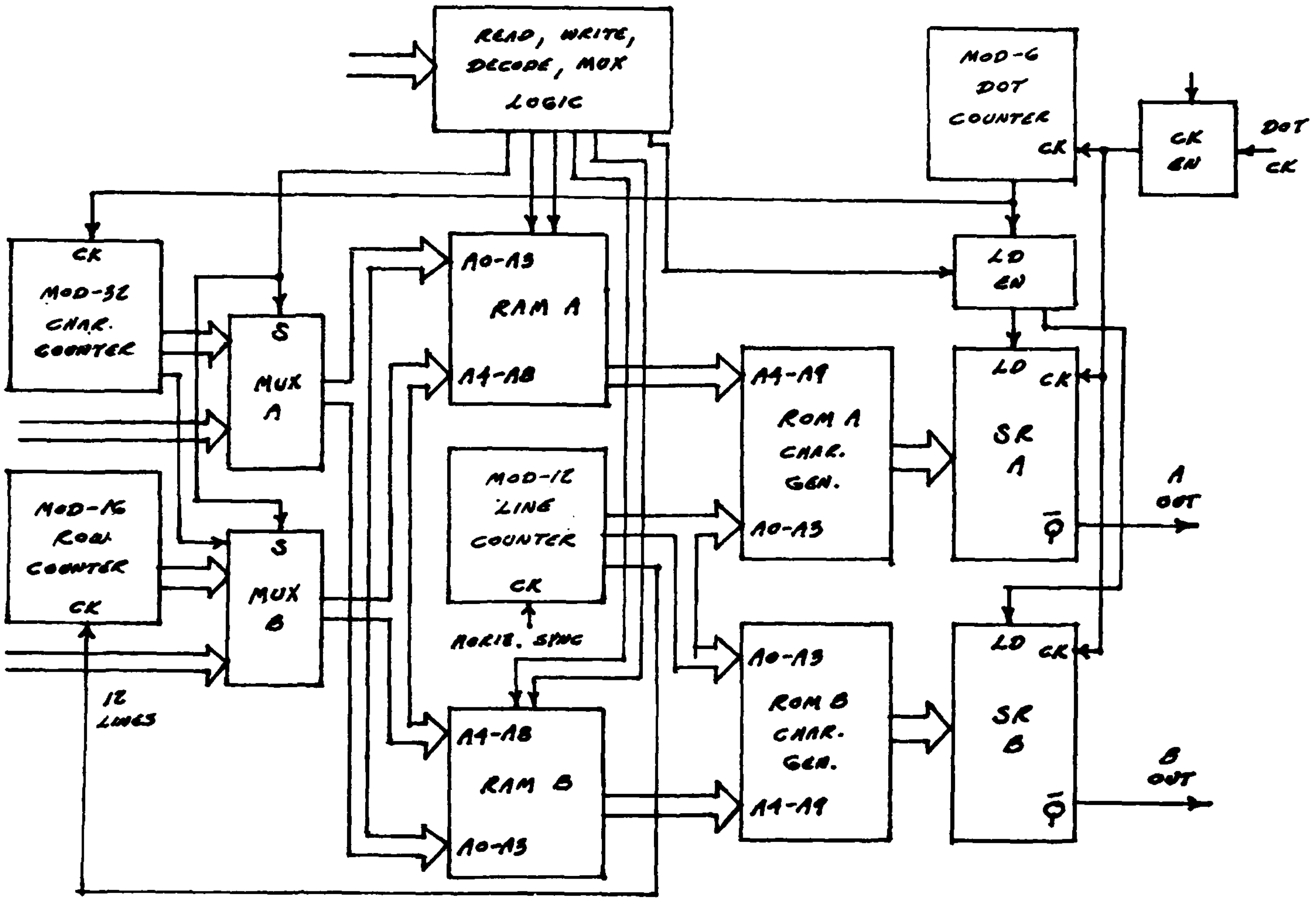

Fig. 2.1. Text Generator Block Diagram

dot and character counters. Also, an external vertical synchronizing pulse is required to start a new data frame. This clears the line counter and row counter. RAM decoding is such that data frames for both output channels are sequenced simultaneously. However, data cannot be stored in RAMS A and B simultaneously. In fact 2 separate 8 bit bidirectional data ports are used to either write data into or read data out of either RAM. Multiplexors (MUXS A and B) which are controlled by the RAM read/write/decode/MUX logic allows RAMS A or B to be addressed by an external system processor rather than by the character and row counters.

2.1. RAM Read/Write/Decode/MUX Logic

A schematic showing the major components of the read/write/decode/MUX logic block of Fig. 2.1 is shown in Fig. 2.2. In order to address the text generator RAM externally, 6 processor address lines (terminals A06-A09, A13, A14) must have coincidence with a hard wired 6 bit address on Bus Comparator U12. Eight additional processor address lines (not shown) are applied through inverting Schmitt triggers (U11, U22) and MUXA and MUXB to the 8 address inputs of each RAM. The two remaining address lines in a 16 line bus are used to decode RAM high and low addresses (terminal A26) and to select either RAM A or RAM B (terminal A25) also through inverting Schmitt triggers (not shown) and MUX U43.

After a particular address in either RAM (say RAM A) is selected, data can be read by setting terminals B18 (read) = 0 and B19 (write) = 1. This provides a logic 1 output at both terminals C12 and C15. This combination is necessary in order to read data out of the bidirectional 8 bit data port for RAM A, consisting of U33 and U34 (not shown). Terminal C12 = 1 is also the read enable condition for RAM A. To write data into RAM A, set terminals B18 (read) = 1 and B19 (write) = 0. Also, FF U72B $\overline{Q}$ output must be at logic 1. This provides a logic 0 output at both terminals C12 and C15. This combination is necessary to write data into the bidirectional data port for RAM A. Terminal C12 = 0 is also the write enable condition for RAM A. Note that if terminals B18 and B19 are both at logic 1 or if bus coincidence does not occur, terminals C15 = 0 and C12 = 1, which puts the bidirectional data port in a high impedance state. This also deselects the external processor address inputs and connects the RAM address inputs to the row and character counters.

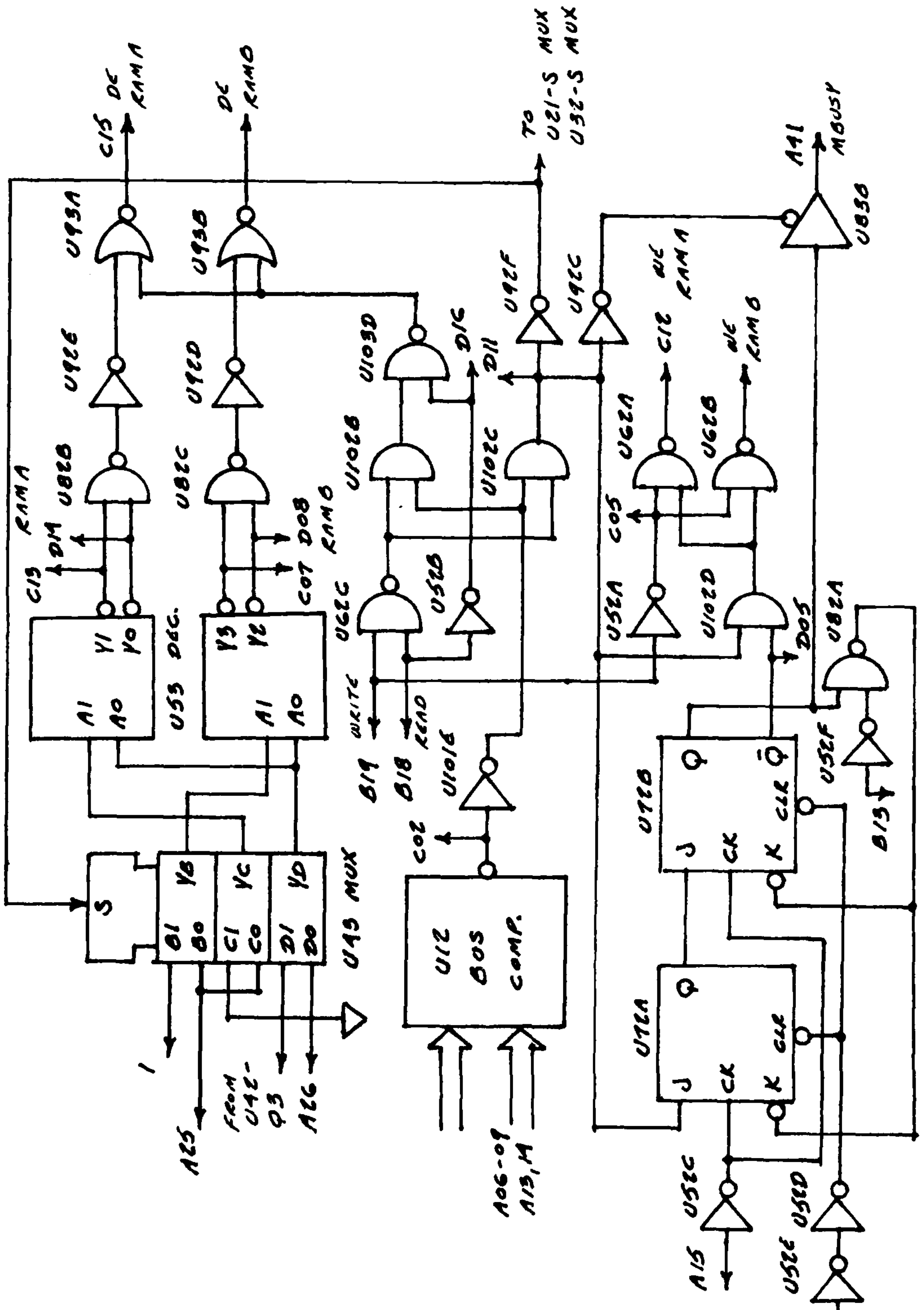

Fig. 2.2. RAM Read/Write/Decode/MUX Logic

The mod-3 counter U72, which is driven by the external system processor clock, is designed to prevent data from accidentally being written into the previous RAM address because of finite RAM address access time. (i.e., data arrives before the RAM address has time to change.) This has no effect on the read operation.

Each RAM consists of four 256 x 4 9112 RAM chips, which are connected as a 512 x 8 RAM. Only 7 bits of each 8 bit word are used by the system. Six of these are used for the ROM character addresses and 1 bit is used to enable the ROM's.

2.2 Character Generator Circuitry

The character generator circuitry includes all of the text generator circuitry except for the RAMS and the RAM read/write/decode/MUX logic. Both for fault diagnosis and descriptive purposes, it is convenient to divide this circuitry into 4 major functional blocks. These are:

1) Channel A Character Generator
2) Channel B Character Generator
3) Horizontal Timing Circuitry
4) Vertical Timing Circuitry

The channel B character generator shown in Fig. 2.3 includes all character generator components that are used in channel B only. This includes the character generator ROM (U84, U85) and shift register U94, as well as inhibiting gates (U103A, U93C). Similarly, the channel A character generator, whose schematic is identical to that for channel B except for part numbers, includes all character generator components that are used in channel A only.

The timing circuitry is used to sequence and control both output channels. Generally speaking, the horizontal timing circuitry shown in Fig. 2.4 includes all of the timing circuitry required to sequence one horizontal line of data plus blanking. This includes the dot clock enable circuitry, dot counter (U71), and character counter (U31 and U41) which have previously been described. NAND gate U61 which inhibits output data from either channel for horizontal and vertical blanking is shown in both timing circuitry schematics for convenience. Note that register U51 provides horizontal blanking after one line of 32 horizontal characters has been sequenced. Programmable counter U24 and FF U23A provide a blank

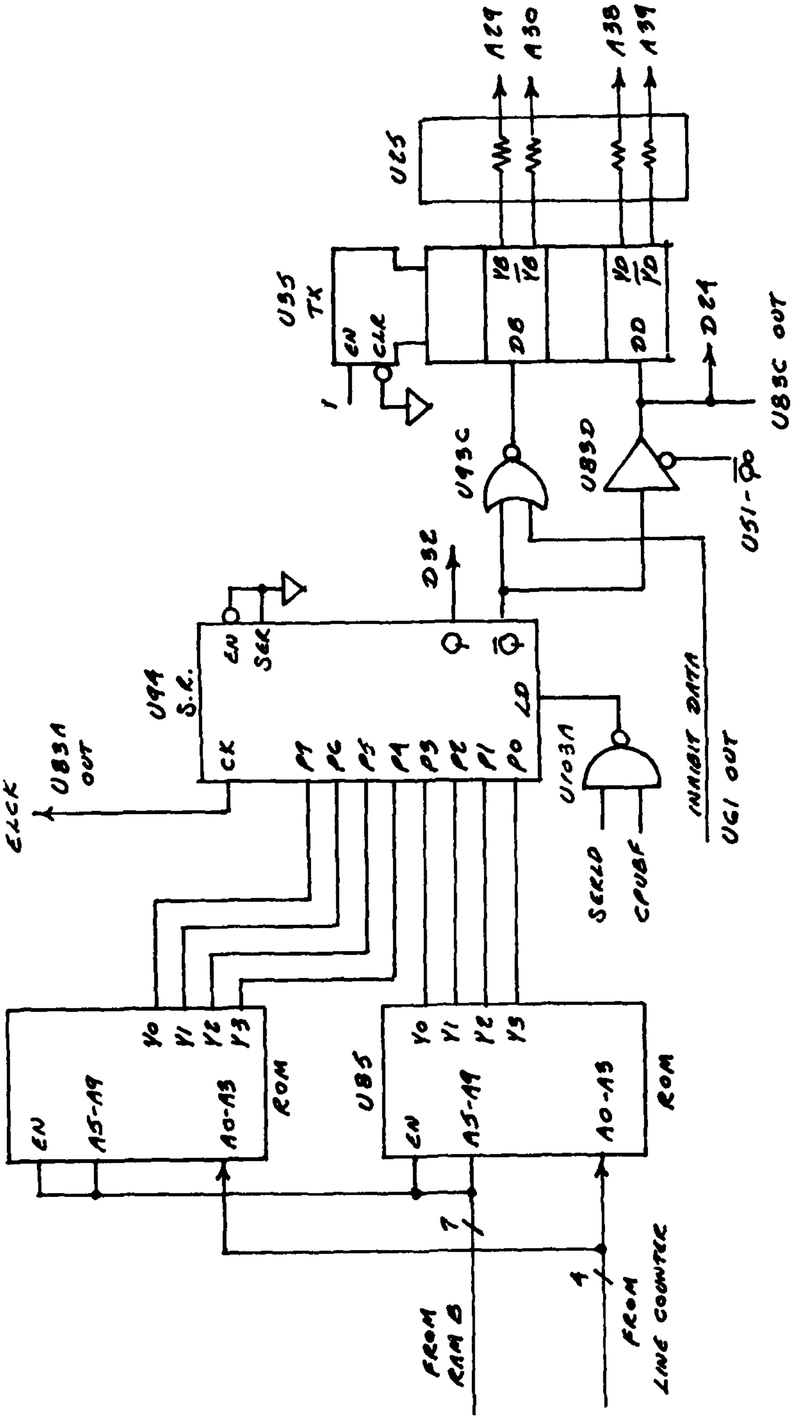

Fig. 2.3. Channel B Character Generator

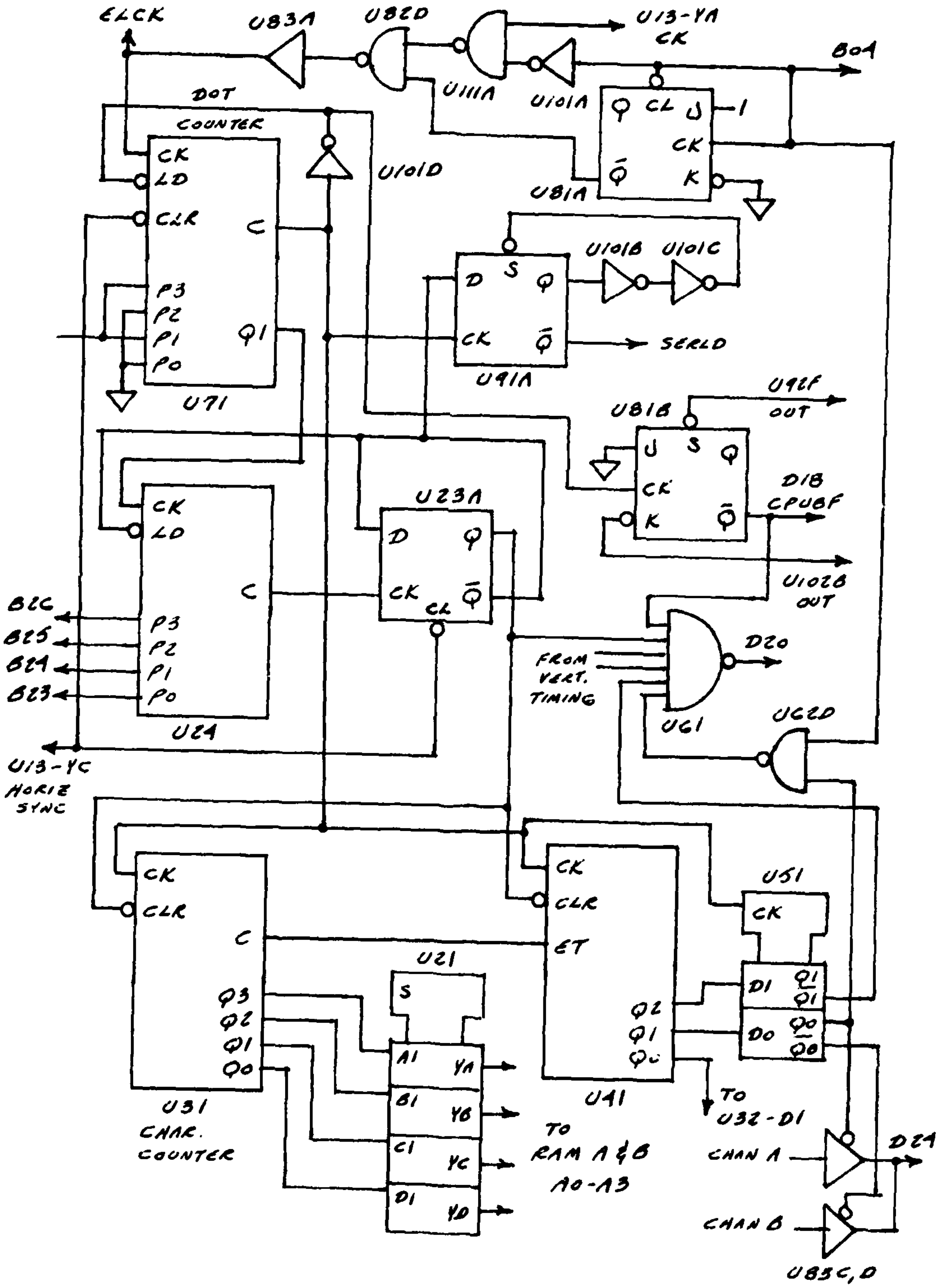

Fig. 2.4. Horizontal Timing Circuitry

horizontal margin following horizontal synchronization. FF U91A acts as a one shot to assynchronously load the output shift registers after 6 dots are counted by the dot counter. FF81B must also be cleared by the dot counter to enable shift register loading. This FF is assynchronously set when RAM is addressed externally.

Components U62D, U82D, U111A, U101A, U81A, and U83C,D and terminals B04 and D24 serve no functional purpose. These have been added by the manufacturer for test purposes. However, since these test points are not used in the fault isolation procedures described in this report, these additional components, which represent a sizeable fraction of the total chips in the horizontal timing circuitry, severely complicate fault isolation in this circuitry. This is one of many examples where obsolete design for test rules, designed to make inefficient fault diagnosis procedures easier, actually make efficient fault diagnosis procedures more difficult.

The vertical timing circuitry shown in Fig. 2.5 includes all of the timing circuitry required to sequence the output data vertically and provide vertical blanking. This inlcudes the line counter (U112, U92B, U111B, and U102A) and row counter (U42) which have previously been described. Note that FF U91B provides vertical blanking after 16 character rows have been sequenced. Programmable counter U15 and FF U23B provide a blank vertical margin of at least one character row after a vertical synchronizing pulse starts a new frame.

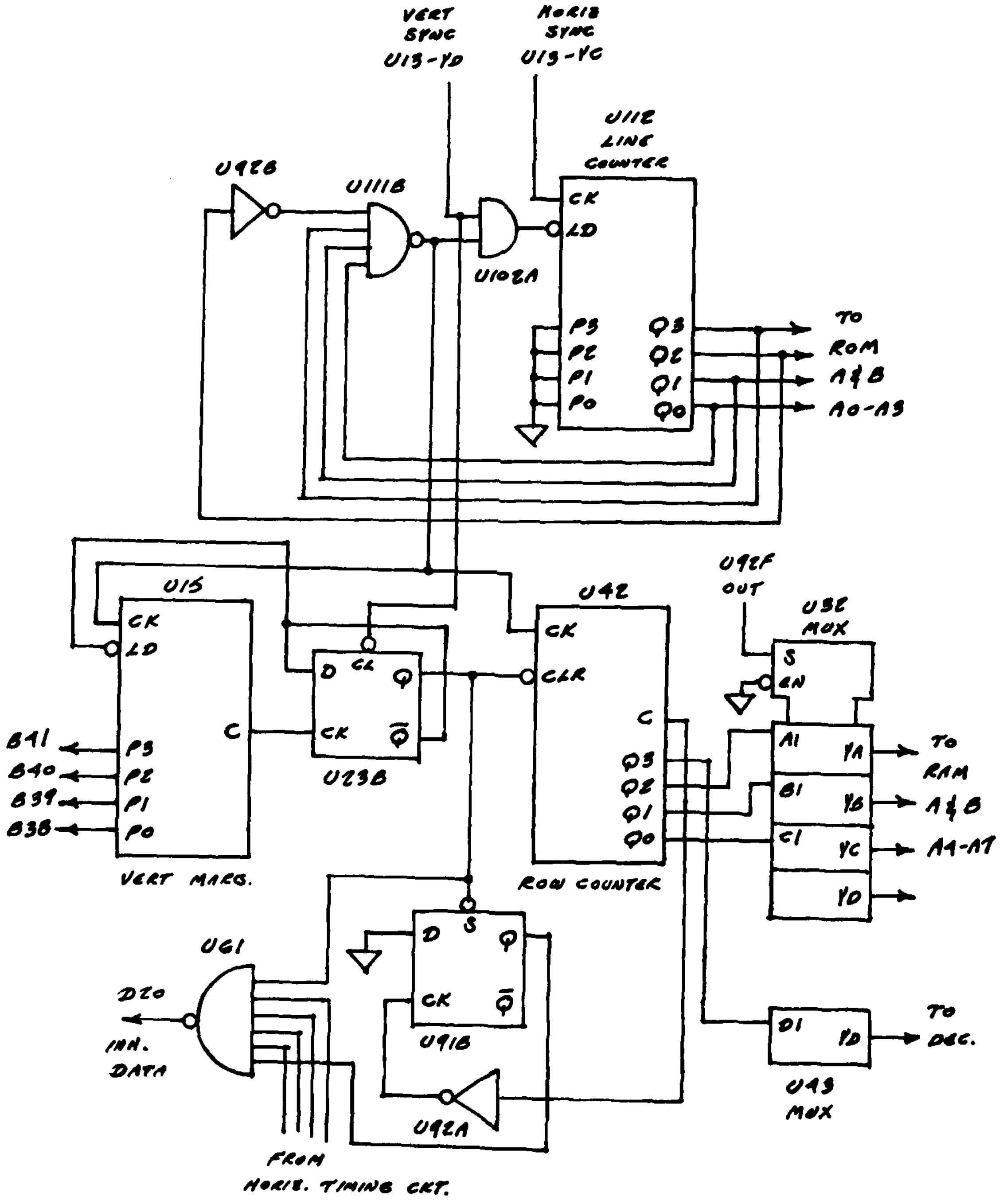

Fig. 2.5. Vertical Timing Circuitry

3.0. SYSTEM FUNCTIONAL FAULT DIAGNOSIS

This section describes the test philosophy and fault diagnosis procedures developed for the text generator. These are designed primarily to minimize the time required for the test processor to detect and isolate faults and also to minimize the number of interfaces between the test processor and the unit under test. The test procedure is sequential in that the test sequence is dependent upon the results of prior tests. This helps to minimize the time required to isolate faults. It is also system functional in that the system is exercised to perform the functions for which it was intended. A fault is defined if the system is unable to perform a function for which it was intended or if it performs a function for which it was not intended.

The first step in the fault diagnosis procedure is to list the major functional tests for the system in the sequence that they must be performed. In the case of the text generator, two major functional tests can be defined.

1) Write character addresses into RAM. Read character addresses stored in RAM.
2) Sequence ROM character generator. Read character data streams at channel outputs.

Defining major system functional tests has several advantages. First, it greatly simplifies input test generation. This is simply chosen to exercise all of the system functions. Note that this implies that path sensitization is being employed. Secondly, it allows a large number of chips to be exercised during a single test, enabling rapid detection of faults. Also, there are no theoretical restrictions on the type of fault that can be detected. Any single or multiple chip fault that causes the system to malfunction will be detected. However, in general multiple chip faults cannot be isolated because most of the fault isolation procedures are based upon the assumption of a single chip fault. (Where a fault symptom cannot possibly be caused by a single chip fault, a double chip fault is assumed). A practical restriction is that transient faults may not be detected if the system is exercised in slow motion for diagnostic purposes.

If more than one system functional test can be defined, then fault detection automatically provides a certain amount of fault isolation. This is because only a certain percentage of chips are involved in any one

TABLE 3-1. SYSTEM FUNCTIONAL TEST FAULT ISOLATION

RAM R/W Chips (17)	Shared Chips (12)	Char. Gen. Chips (22)
U74 RAM	U21 MUX	U71 Counter
U63 RAM	U32 MUX	U112 Counter
U73 RAM	U43 MUX	U15 Counter
U64 RAM	U11 Schmitt T.	U24 Counter
U65 RAM	U83 Tri-State Buf.	U31 Counter
U75 RAM	U101 Inv.	U41 Counter
U55 RAM	U92 Inv.	U42 Counter
U45 RAM	U102 AND	U84 PROM
U12 BUS Comp.	U62 NAND	U85 PROM
U33 BUS Driv/Rec.	U82 NAND	U104 PROM
U34 BUS Driv/Rec.	U103 NAND	U113 PROM
U54 BUS Driv/Rec.	U93 NOR	U94 S.R.
U44 BUS Driv/Rec.		U114 S.R.
U53 Decoder		U23 D FF
U72 FF		U91 D FF
U52 Inv.		U81 J-K FF
U22 Schmitt Trig.		U13 Rec.
		U35 Tx
		U51 Quad. Reg.
		U61 NAND
		U111 NAND
		U25 Res.

blocks are in error. Error code 00 indicates that no data blocks are in error.

The next step is to partition the chips listed in columns 1 and 2 of Table 3-1 according to the error codes that can be produced by a single chip error. For example, error code OF implies that the error exists in one of the read/write/decode/MUX logic chips which affect all RAM. Thus all of these chips are included in this fault category. If a single chip fault cannot produce a particular error code, then partitioning is based upon a double chip fault assumption. This is the case for error codes 07, OB, OD, and OE. The results of this partitioning for the RAM write/read test is shown in Table 3-2 for single chip faults and Table 3-3 for double chip faults.

Note that in the best case (error codes 05 and OA), fault isolation to within 3.9% (2 of 51) of the systems chips is possible. For the case of error codes 01, 02, 04, and 08, which include any single RAM chip fault, fault isolation to within 5.9% (3 of 51) of the systems chips is possible. This is significant since RAM chips are probably the most unreliable in the system. In the worst case (error code OF), fault isolation to within 25.5% (13 of 51) of the system chips occurs. This latter percentage could be reduced to 19.6% (10 of 51) if both NAND gates (U62C and U103D) were located on the same chip and if all inverters in this error category were located on one chip.

It is significant that this degree of fault isolation is achieved by performing a single functional test, and the only test points used are the system primary inputs and outputs which are located on the system bus. In the case of single chip RAM faults, additional fault isolation to within a single chip is possible without moving off the system bus. This is important in built-in testing (BIT) where the use of internal test points may be prohibited. It illustrates that BIT on the system bus need not be limited to a simple go-no go decision.

However, it should not be inferred that a high degree of fault isolation can always be achieved without observing some internal test points. To illustrate this, consider Table 3-4, which partitions the chips exercised during the character generator test according to whether a data frame error exists in the channel A output only, the channel B output only, or in both channel outputs. In the latter case a single chip fault can functional test. For diagnostic purposes it is convenient to partition the system chips into functional test categories. This partitioning is illustrated in Table 3-1 for the text generator. The first group of 17 chips are those which are used exclusively in the RAM write/read test and the last group of 22 chips are those used exclusively in the character generator test. The remaining group of 12 chips are used in both functional tests. Note that RAM chips are included in the first group even though RAM is used to sequence the characters in the character generator test. This is because RAM faults will be detected in the RAM write/read test, which precedes the character generator test.

Thus an error detected during the RAM write/read test automatically isolates the fault to within 57% (29 of 51) of the system chips, whereas an error in the character generator test isolates the fault to within 67% (34 of 51) of the system chips. Note that this degree of isolation is independent of whether a single or multiple chip fault exists.

The degree of fault isolation provided by the system functional tests can be improved significantly if the output data is divided up into functional data blocks. Instead of simply noting that an error occurs during a functional test, the data block or group of data blocks in which an error occurs is also noted. To minimize sequencing time, the program moves to the next block as soon as a single error is detected in the preceding data block.

To illustrate this, consider the RAM write/read test. RAM data can be divided into 4 blocks as follows.

01) RAM A Low Addresses

02) RAM A High Addresses

04) RAM B Low Addresses

08) RAM B High Addresses

The hex numbers 01, 02, 04, and 08 are error codes used to designate 4 of 16 possible error categories that can occur during the RAM write/read test. If an error exists in only a single data block, then the applicable error code is stored in a memory location to enable the test processor to remember where the error occurred and to determine what test must be performed next. If an error exists in more than one data block, the applicable error code is obtained by adding the hex numbers for each faulty data block. For example, OF = 01 + 02 + 04 + 08 indicates that all data

TABLE 3-2. RAM W/R TEST SINGLE CHIP FAULT ISOLATION

01) RAM A Low Addresses (3)	04) RAM B Low Addresses (3)
U73 RAM	U55 RAM
U64 RAM	U45 RAM
U53A Dec.	U53B Dec.
02) RAM A High Addresses (3)	08) RAM B High Addresses (3)
U74 RAM	U65 RAM
U63 RAM	U75 RAM
U53A Dec.	U53B Dec.
05) RAMS A & B Lower Addresses (2)	0A) RAMS A & B High Addresses (2)
U43 MUX	U43 MUX
U22 Schmitt T.	U22 Schmitt T.
03) RAM A (9)	0C) RAM B (9)
U62A NAND	U62B NAND
U93A NOR	U93B NOR
U92E Inv.	U92D Inv.
U82B NAND	U82C NAND
U53A Dec.	U53B Dec.
U43 MUX	U43 MUX
U22A Schmitt Trig.	U22A Schmitt Trig.
U33 Buf.	U54 Buf.
U34 Buf.	U44 Buf.

OF) RAM A & B (13)		
U12 BUS Comp.	U92F Inv.	U43 MUX
U101E Inv.	U72 F.F.	U22 Schmitt Trig.
U62C NAND	U103D NAND	U11 Schmitt Trig.
U102B,C,D AND	U21 MUX	
U52A,B,D,E Inv.	U32 MUX	

TABLE 3-3. RAM W/R TEST DOUBLE CHIP FAULT ISOLATION

06)	RAM A_H & RAM B_L (5)	09)	RAM A_L & RAM B_H (5)
	U74 RAM		U73 RAM
	U63 RAM		U64 RAM
	U55 RAM		U65 RAM
	U45 RAM		U75 RAM
	U53 Dec.		U53 Dec.
07)	RAM A & RAM B_L (11)	OB)	RAM A & RAM B_H (11)
	U55 RAM		U65 RAM
	U45 RAM		U75 RAM
	U53 Dec.		U53 Dec.
	U62A NAND		U62A NAND
	U93A NOR		U93A NOR
	U92E Inv.		U92E Inv.
	U82B NAND		U82B NAND
	U43 MUX		U43 MUX
	U22A Schmitt Trig.		U22A Schmitt Trig.
	U33 Buf.		U33 Buf.
	U34 Buf.		U34 Buf.
OD)	RAM B & RAM A_L (11)	OE)	RAM B & RAM A_H (11)
	U73 RAM		U74 RAM
	U64 RAM		U64 RAM
	U53 Dec.		U53 Dec.
	U62B NAND		U62B NAND
	U93B NOR		U93B NOR
	U92D Inv.		U92D Inv.
	U82C NAND		U82C NAND
	U43 MUX		U43 MUX
	U22A Schmitt Trig.		U22A Schmitt Trig.
	U54 Buf.		U54 Buf.
	U44 Buf.		U44 Buf.

TABLE 3-4. CHARACTER GENERATOR TEST FAULT ISOLATION

Channel A (7)	Channel B (7)	Both Channels (25)
U104 PROM	U84 PROM	U71 Counter
U113 PROM	U85 PROM	U112 Counter
U114 S.R.	U94 S.R.	U15 Counter
U93D NOR	U93C NOR	U24 Counter
U103B NAND	U103A NAND	*U31 Counter
U35 TX	U35 TX	U41 Counter
U25 Res.	U25 Res.	*U42 Counter
		*U21 MUX
		*U32 MUX
		*U43 MUX
		U23 D FF
		U91 D FF
		U81 J-K FF
		U51 Quad D Reg.
		U13 Rec.
		U35 TX
		U83A T-S Buf.
		U61 NAND
		U111 NAND
		U62D NAND
		U82D NAND
		U101 Inv.
		*U92 Inv.
		U11 Schmitt Trig.
		*U102A AND

exist in any of 25 chips. Next consider the worst case situation where all data is inhibited at the output of both channels. (i.e., the entire data frame consists of logic "0"s). Eighteen of the 25 chips (those not marked with an asterisk) can cause such an inhibiting fault. Obviously, further fault isolation is impossible without observing data at some internal test point.

Note that although there are actually 53 rather than 51 chips in the text generator module, two of these are integrated pull up resistors. In this report faults in either integrated or discrete pull up resistors and decoupling capacitors are not considered. Generally speaking such faults would have to be catastrophic to have a significant effect on the system.

3.1. RAM Write/Read Fault Diagnosis Procedures

This section describes the detailed RAM write/read fault diagnosis procedures for the text generator. First consider the test data blocks that are written into RAM at the beginning of this test. Since one RAM data bit is not used in the system, this bit should be masked off before reading RAM data. A second RAM data bit is used only to enable PROM. This bit must be set to logic 0. Next consider the following practical restrictions on the remaining 6 character address data bits.

1) Every RAM data block must include all 64 character addresses. This is necessary to insure that the RAM completely exercises the PROM during the character generator test.
2) Every character in each 32 character row must be distinct. Otherwise a fault consisting of the character counter or its associated MUX pointing to a wrong character may go undetected.
3) Every character in each 16 character column must be distinct. Otherwise, a fault consisting of the row counter or its associated MUX pointing to a wrong row may go undetected.

Note that these restrictions will also prevent many internal RAM decoder faults from going undetected.

It should be noted that writing a single block of data into RAM and reading this data is not a complete test of RAM. Consequently, the following procedure is recommended. After writing one data block into RAM and successfully reading and sequencing the PROM with this data, the 6 character address bits in each RAM word are inverted and the RAM read test is repeated. This insures that each significant memory cell is able to store a 1 or a 0.

One possible 32 x 16 RAM test data array that satisfies the above restrictions is shown in Table 3-5. Here the character addresses stored in RAM are written in hex. The array is arranged in the same manner that the corresponding characters are displayed on a CRT. For convenience, the same test data array is stored in both RAM's A and B, since these have separate data ports.

Next consider the initial RAM R/W fault diagnosis flow diagram shown in Fig. 3.1. As discussed previously in Section 3.0, this partitions the RAM data into 4 blocks and stores an error code in memory location EO according to which data blocks are in error. Referring to Table 3-2, since error codes 05 and OA isolate the fault to within two chips, no additional fault isolation is performed. Also, no additional fault isolation is performed for the multiple fault error codes listed in Table 3-3.

Next consider error codes 01, 02, 04, and 08. These occur when an error occurs in only a single data block. Each data block is the data stored in two 256 x 4 RAM chips which are simultaneously selected by the decoder. The 4 LSB of each data word is stored in one RAM chip and the 3 MSB are stored in the other. Hence fault isolation to a single chip is possible by noting whether the error occurs in the 4 LSB or the 3 MSB of a data word or in both. An error in both indicates that the single chip fault is in decoder U53. The flow diagram that performs this fault isolation for error code 01 is shown in Fig. 3.2. Here, error codes 11, 21, and 31 indicate which chip is faulty. Similar fault isolation flow diagrams can be drawn for fault codes 02, 04, and 08.

The only observation test points that have been used in the previous tests are the two 8 bit bidirectional RAM data ports which are on the system bus. System bus terminal A41 (memory busy) must also be observed. Since error code OF indicates that all RAM data blocks are in error, additional fault isolation using only these observation points, if not impossible, is certainly impractical. Consequently, the remaining RAM R/W fault isolation procedures utilize 7 additional internal observation test points which are on the test bus. These are the 4 decoder outputs C07, D08, C13, and D14, the MUX select signal, D11, and the RAM A read/write outputs, C12 and C15. Test bus terminals, C02, C05, D05, and D16 are not used. Use of these terminals would require an additional 8 bit port interface between the system and the test processor, while providing only a slight improvement

TABLE 3-5. 32 x 16 RAM TEST DATA ARRAY

01	02	03	04	05	06	07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20
21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00
02	03	04	05	06	07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21
22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00	01
03	04	05	06	07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22
23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00	01	02
04	05	06	07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23
24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00	01	02	03
05	06	07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24
25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00	01	02	03	04
06	07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25
26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00	01	02	03	04	05
07	08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26
27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3E	3E	3F	00	01	02	03	04	05	06
08	09	OA	OB	OC	OD	OE	OF	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	00	01	02	03	04	05	06	07

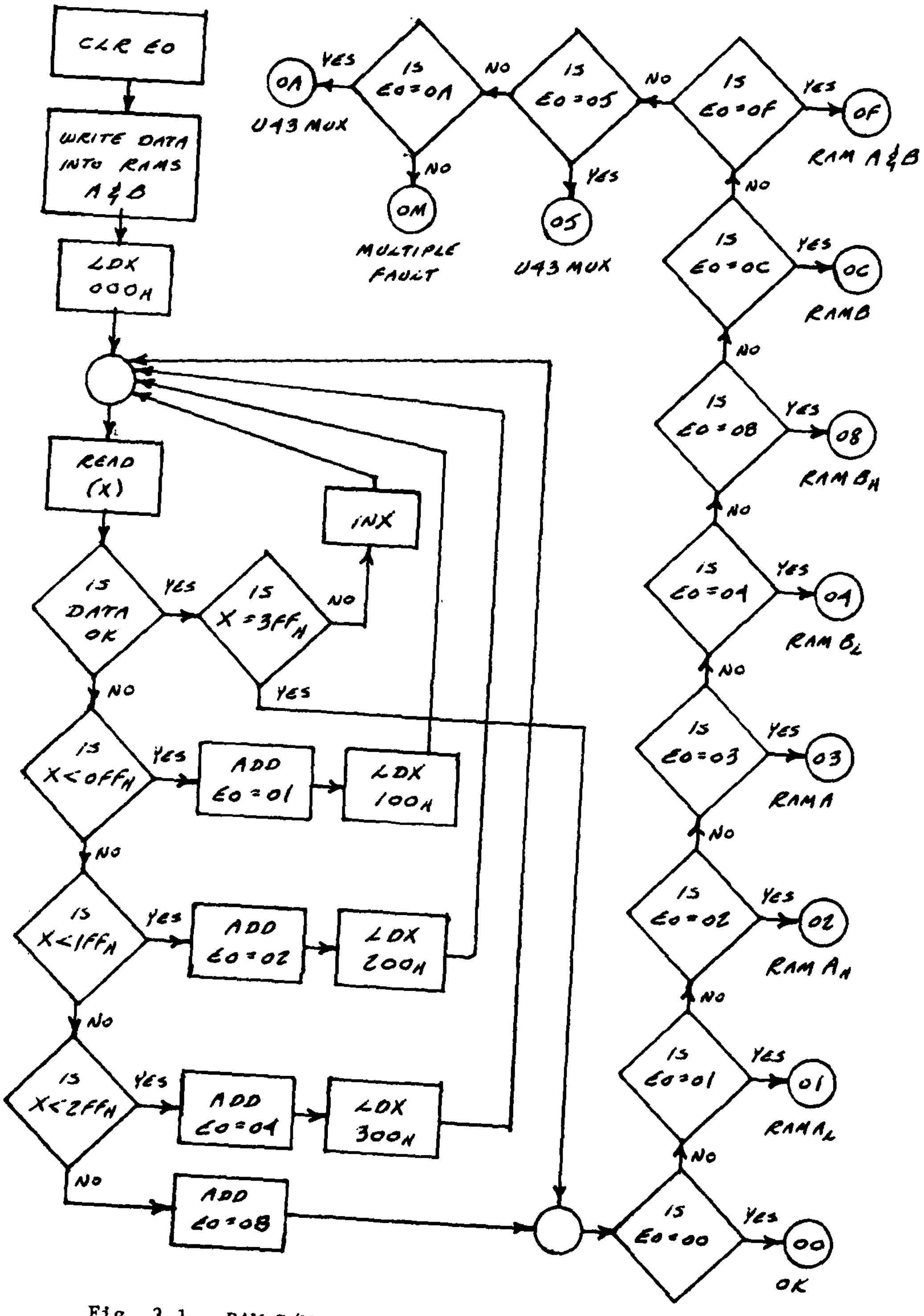

Fig. 3.1. RAM R/W Initial Fault Diagnosis

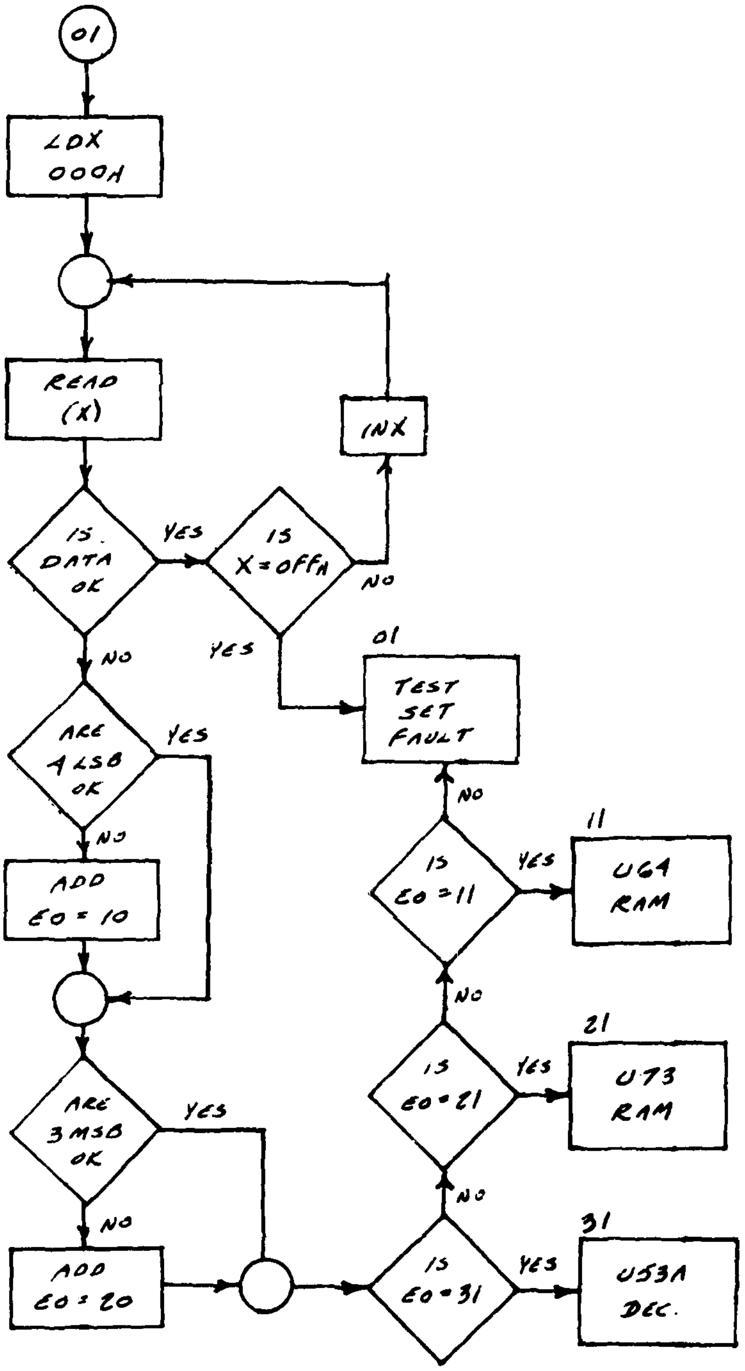

Fig. 3.2. RAM A_L Fault Isolation Subroutine

in the degree of fault isolation obtained. Also, it is time consuming for the test processor to examine more than 8 data bits in a particular test subroutine.

Next co:sider the fault isolation subroutine for error code OF shown in Fig. 3.3. This tests the RAM read/write/decode/MUX logic shown in Fig. 2.2, using standard combinational logic fault isolation techniques, except that it only tests for single faults that can cause an error in every data block. This test produces 10 possible error codes, OF through 9F. Three of these, IF, 6F, and 9F, isolate the fault to a single chip. Three of these, 3F, 7F, and 8F, isolate the fault to 1 of 2 chips. Two of these, 4F and 5F, isolate the fault to 1 of 3 chips. Finally two of these, OF and 2F, isolate the fault to 1 of 4 chips.

The fault isolation subroutine for error code 03 is shown in Fig. 3.4. This tests for single faults in the RAM read/write/decode/MUX logic shown in Fig. 2.2, that can cause errors in both RAM A data blocks. This test produces 7 possible error codes, 03 through 63. Four of these 23, 33, 53, and 63, isolate the fault to a single chip. Two of these, 04 and 43, isolate the fault to 1 of 2 chips, and one of these, 13, isolates the fault to 1 of 3 chips.

The fault isolation subroutine for error code OC is shown in Fig. 3.5. This tests for single faults in the RAM B decoder U53B and the U43 MUX. In the event that a fault does not exist in either of these components output, error code OC indicates that the fault may exist in any 1 of 6 chips. The reason that fault isolation is not as good as for the RAM A fault isolation subroutine is that the RAM B read/write outputs are not available as test points.

Next consider error code 00 which indicates that all RAM data is correct. A fault can still exist in the RAM read/write logic which allows data to be written into or read out of RAM when it is not supposed to. The test subroutine shown in Fig. 3.6 is designed to test for such faults. This test produces 5 possible error codes, 10 through 50, which isolate such faults (if they exist) to within a maximum of 2 chips. Note that this test assumes that bus comparator U12 has a hard wired decode address of all 1's. It can be easily modified to accommodate a different hard wired address.

If the system passes the previous test, the mod-3 counter test shown in Fig. 3.7 is performed. If the system passes this test, the character generator test begins.

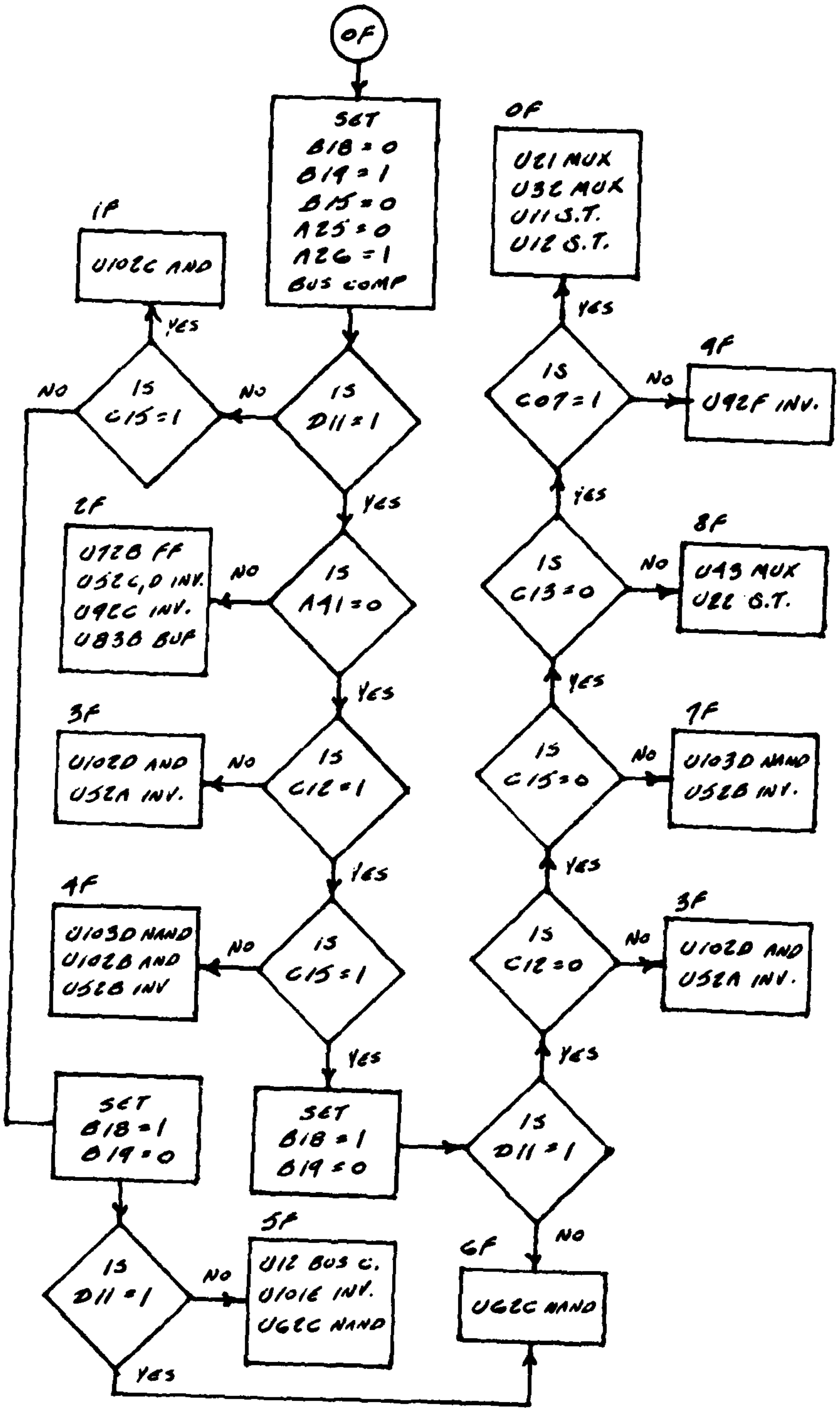

Fig. 3.3. RAM A & B Fault Isolation Subroutine

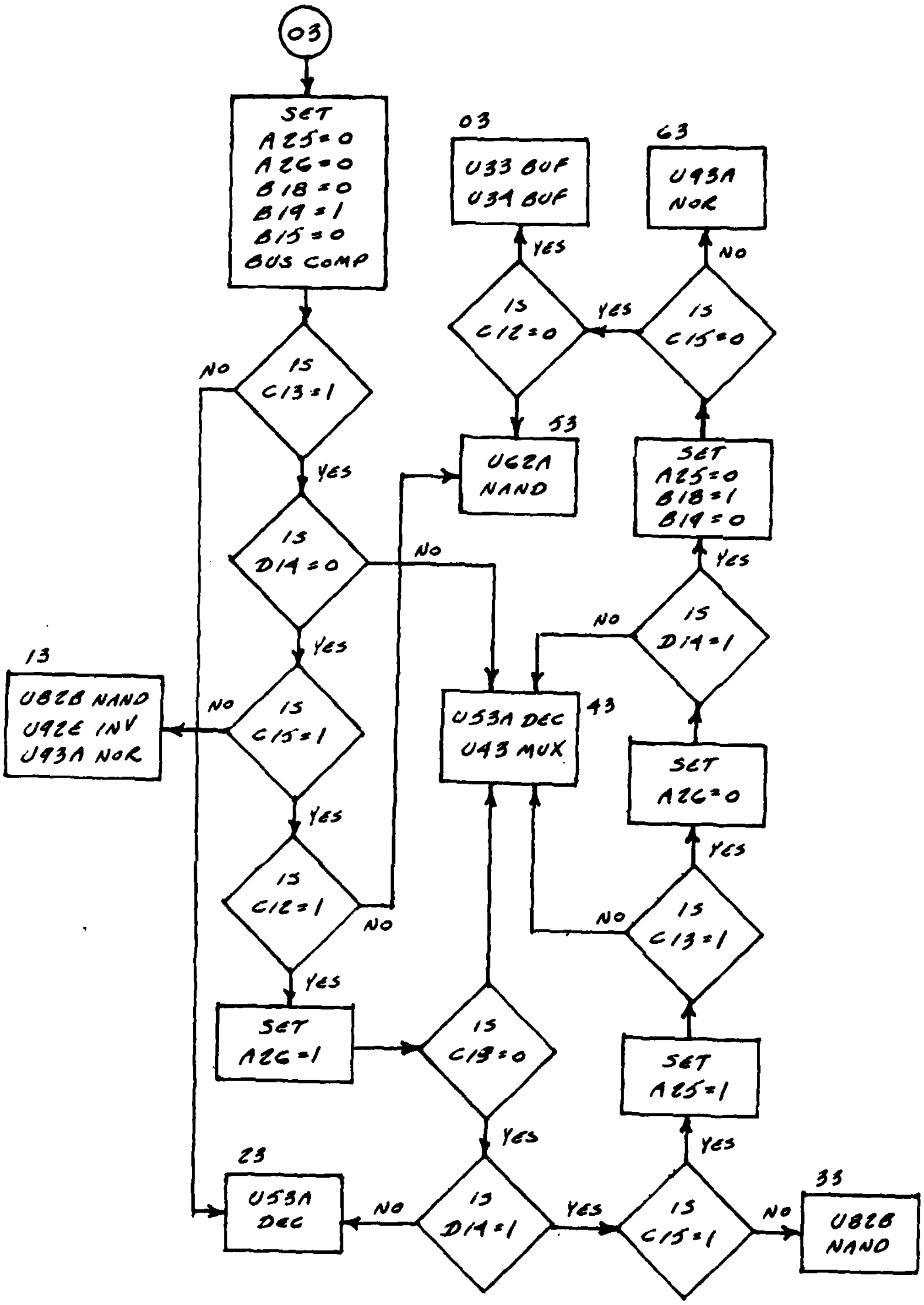

Fig. 3.4. RAM A Fault Isolation Subroutine

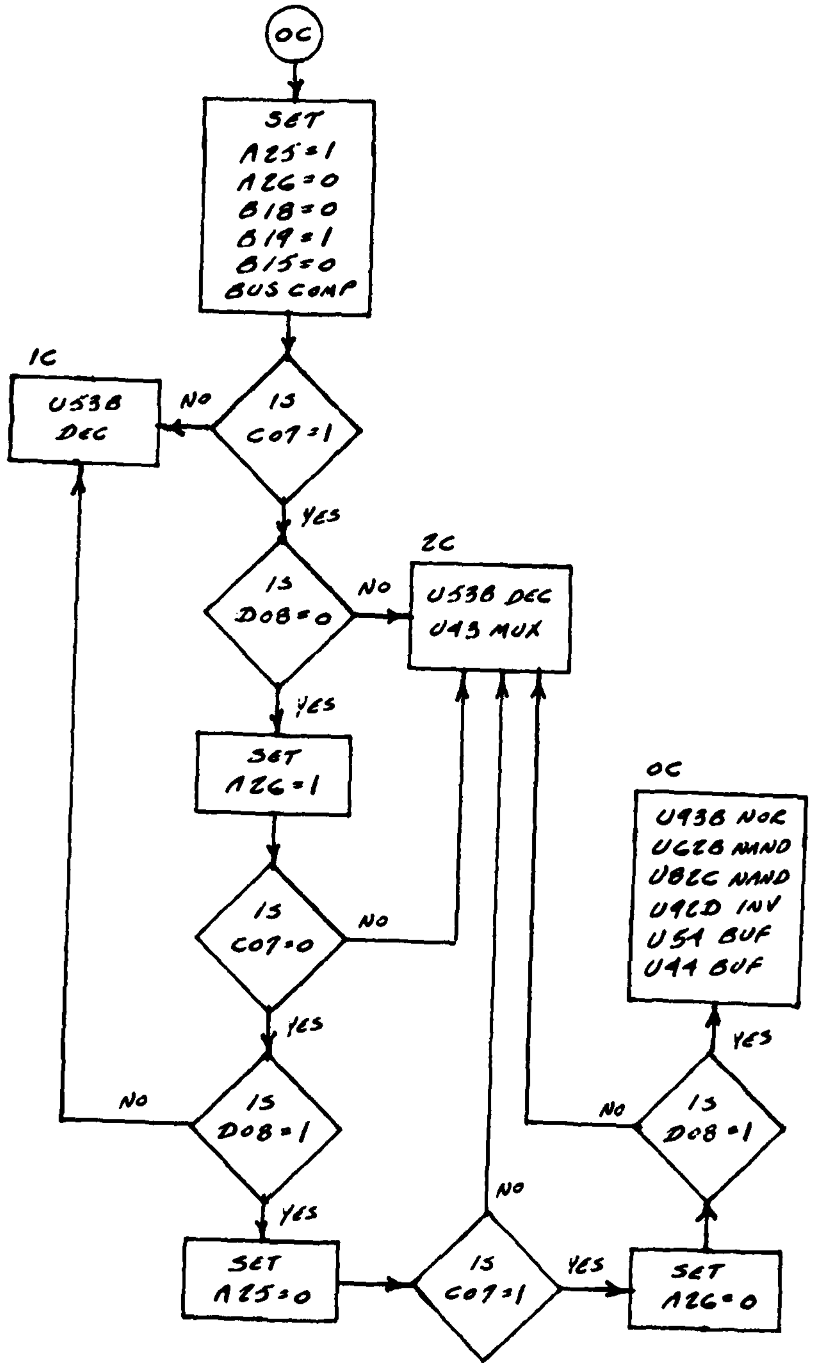

Fig. 3.5. RAM B Fault Isolation Subroutine

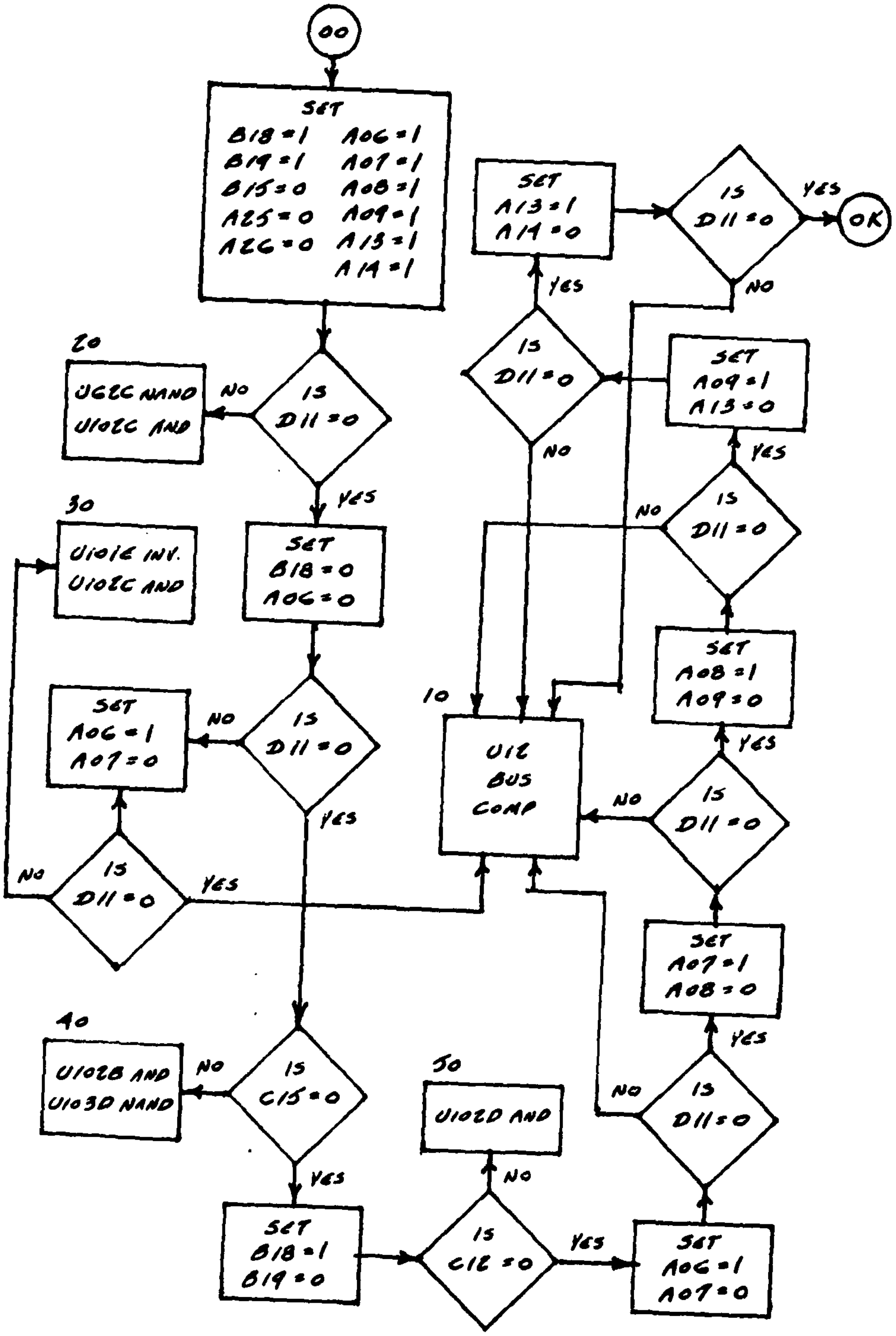

Fig. 3.6. RAM Data OK Subroutine

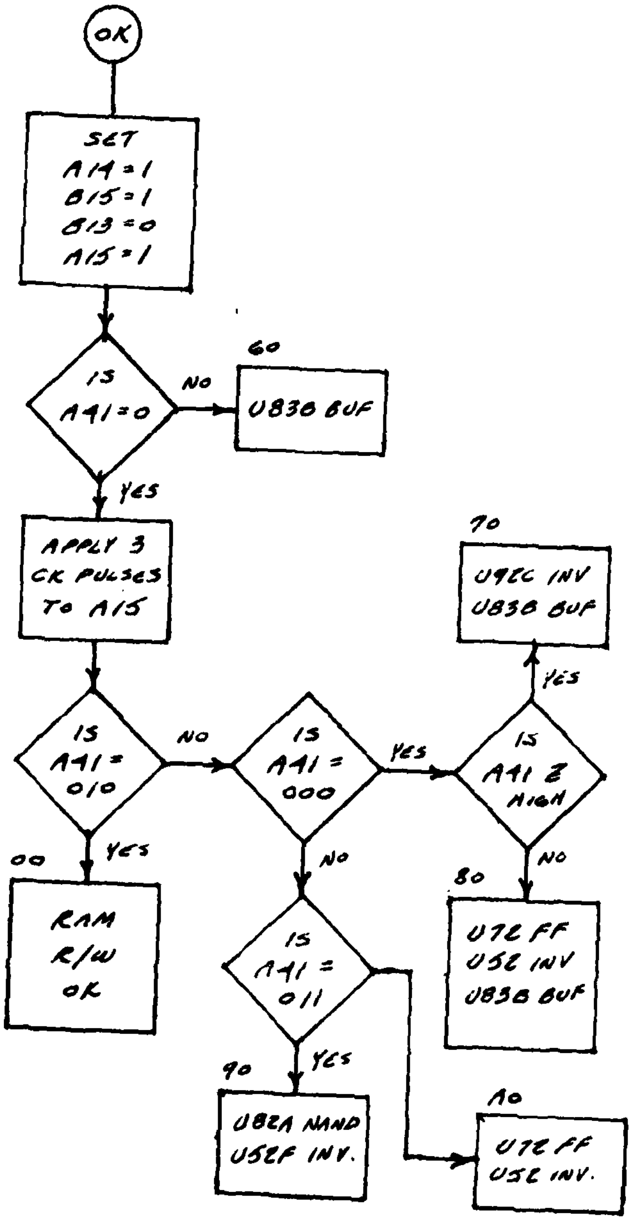

Fig. 3.7. Mod-3 Counter Test

3.2. Character Generator Fault Diagnosis Procedures

This section describes the detailed character generator fault diagnosis procedures. The initial fault diagnosis procedure is designed to detect and classify errors in an entire frame of data at the channel outputs. This consists of a 32 x 16 array of characters as shown in Table 3.5. Each character is 6 bits wide and 12 lines long including spaces between characters. Thus an entire data frame consists of 32 x 16 x 6 x 12 = 37, 164 data bits, exclusive of horizontal and vertical margins and blanking. Consequently, it is highly desirable to employ a proven, reliable, data compression technique to minimize test processing time. For this purpose, a pseudo-random sequence generator, consisting of a 16 bit shift register with the mod-2 sum of bits 7, 9, 12, and 16 fed back into its input, is proposed as an interface between the test processor and the system. This is identical to the circuit used in the HP 5004A Signature Analyzer [1]. Thus the correct signature for a particular array of characters can be determined experimentally.

Although the signature analyzer compresses a data stream of tens of thousands of bits into a 16 bit signature, it will always identify a correct data stream and the probability that an incorrect non-zero data stream will go undetected is only $1/2^{16}$. Also, if the signature analyzer is initially cleared, a data stream consisting of all zeroes will produce a signature of 0000_H. On the other hand, a data stream containing one or more ones cannot produce a signature of 0000_H. Thus it is possible to distinguish between a non-zero data stream that is incorrect and a data stream that is inhibited. This is important since there are many single chip faults that can inhibit an entire data frame.

Using a hardware signature analyzer that is sequenced by the same external dot clock that is used to sequence the system allows the system to be sequenced at its normal speed (e.g., 30 frames/sec.). This not only minimizes test time but also makes it possible to detect transient errors in the data stream. The signature analyzer is multiplexed to sequentially examine both the channel A and channel B outputs, and if necessary, either one of two internal test points (C32 and D32), which are the outputs of the channel A and channel B shift registers, respectively. These 4 test points are the only ones used for signature analysis in the succeeding test procedures.

Before considering the detailed fault diagnosis procedures, it is necessary to clarify some terminology. In the succeeding discussion, row 1 through row 16 of a data frame refer to row 1 through 16 of a data array such as that shown in Table 3-5. Similarly, characters 1 through 32 refer to characters 1 through 32 of this array. On the other hand, row 17 refers to the blank row (or rows) that follow row 16 and character 33 refers to the blank data that follows character 32. It is necessary to include these blank rows and characters in a complete data frame to insure that the horizontal and vertical blanking circuitry is functioning properly. In addition, assuming that counter U15 of Fig. 2.5 is hard wired to load 1110, one blank row of data must precede row 1. This is referred to as row 0. Similarly, the blank space, consisting of at least 16 dots, preceding each line of character 1 is referred to as character 0.

Next consider the initial fault diagnosis flow diagram shown in Fig. 3.8. This is designed to detect errors in a data frame of the channel A and channel B outputs and classify these errors according to the following error codes which are stored in memory location E1.

00) No Error

01) Channel A Only

02) Channel B Only

03) Both Channels, Signature $\neq 0000_H$

04) Both Channels, Signature $= 0000_H$

05) Both Channels, A Sig. $\neq$ B Sig. (Multiple Fault)

Note that error code 05 indicates a multiple fault since the same data array is stored in both RAM's A and B. No additional fault isolation is attempted for this case. Also, no additional fault diagnosis is necessary for fault code 00.

Consider the fault isolation subroutine for error code 02 shown in Fig. 3.9. This error code indicates that the error exists in the channel B output only. This implies a single chip fault in 1 of 7 chips shown in Fig. 2.3. The procedure produces 8 error codes, 12 through 82, which isolate the fault to within at most 3 chips in 2 cases, 2 chips in 5 cases, and 1 chip in 1 case. The only observation test points used are the channel B output and D32 in Fig. 2.3. Fault isolation is accomplished by examining only single bytes of data. A similar fault isolation subroutine can be used for an error in the channel A output only, which is indicated by error code 01.

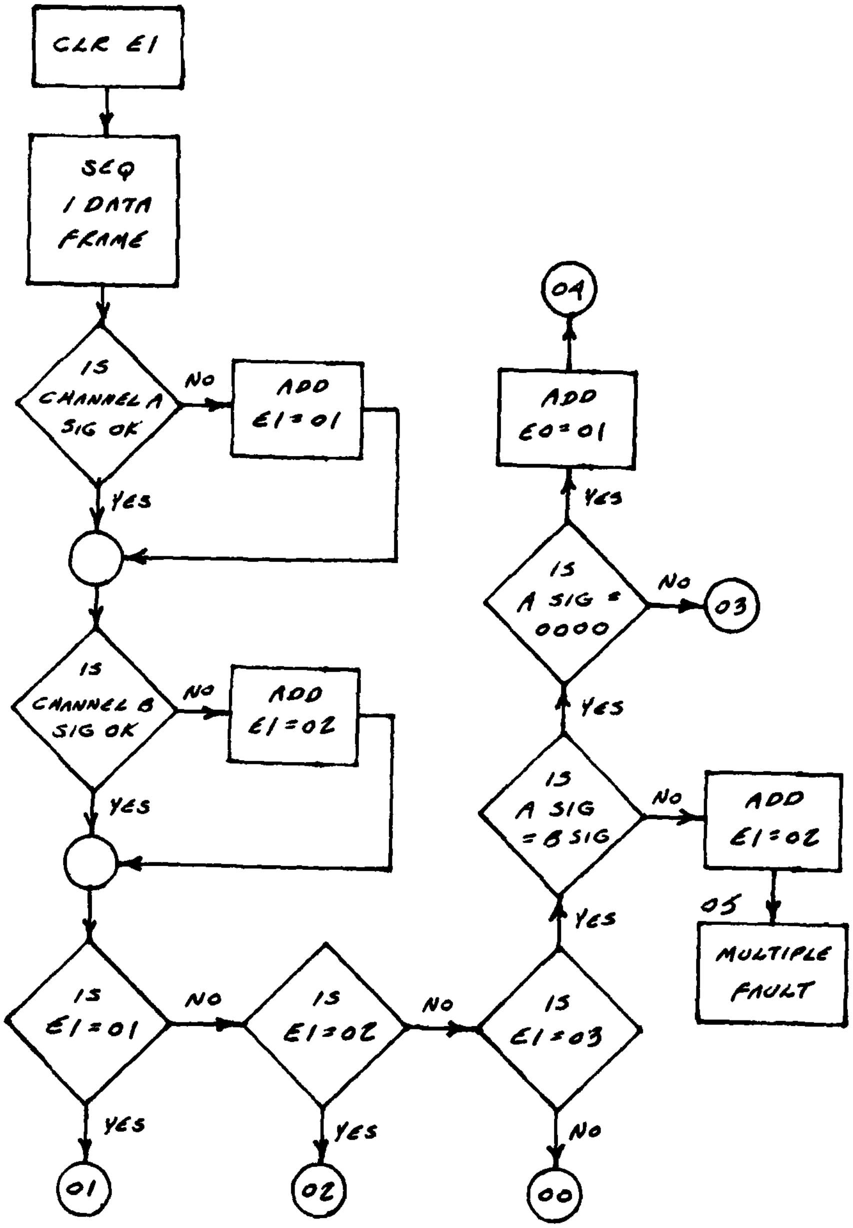

Fig. 3.8. Char. Gen. Initial Fault Diagnosis

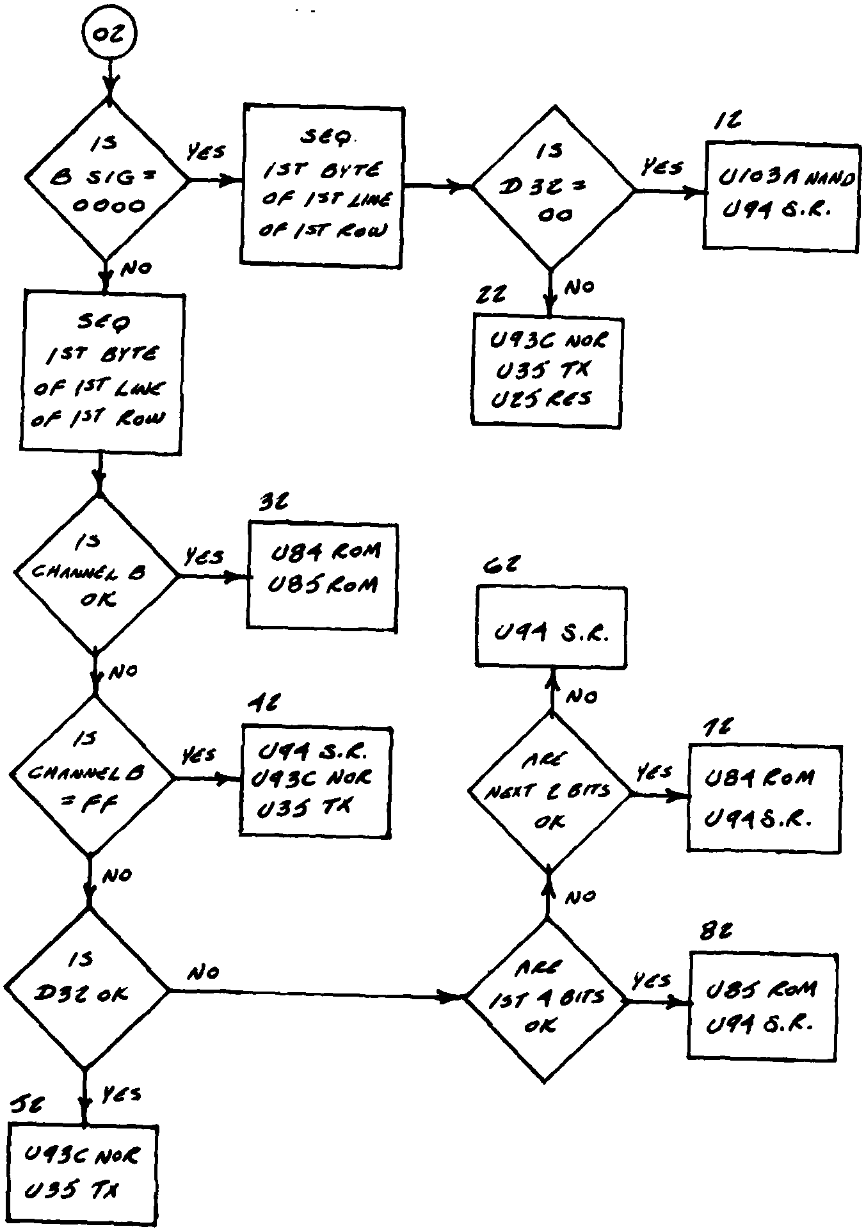

Fig. 3.9. Channel B Char. Gen. Fault Isolation Subroutine

Next consider the fault isolation subroutine for error code 03 shown in Figs. 3.10, 3.11, and 3.12. Error code 03 indicates that both channels are in error but the data is not all zeroes. This implies a single chip fault in the timing circuitry of Figs. 2.4 and 2.5. Figure 3.10 is essentially a test of the vertical timing circuitry of Fig. 2.5 and Figs. 3.11 and 3.12 is a test of the horizontal timing circuitry of Fig. 2.6. The procedure produces 18 error codes, 03 through 73, 06 through 66, 07 through 27, which isolate the fault to within at most 5 chips in 1 case, 4 chips in 3 cases, 3 chips in 2 cases, 2 chips in 8 cases, and 1 chip in 2 cases. In addition, output error code 03 denotes a multiple fault in the vertical timing circuitry, and error code 06 denotes a test fault. The entire procedure uses only a single observation test point, the channel B output. Fault isolation is accomplished by examining small data blocks such as a single horizontal line or a single character.

Finally consider the fault isolation subroutine for error code 04 shown in Fig. 3.13. Error code 04 indicates that the entire output data frame for both channels consists of all zeroes. As noted in Table 3-4, 18 single chip faults can cause this symptom. If the same symptom does not appear for the 1st line of data row 1 at test point D32, then the fault is in 1 of 8 chips which are bypassed by this test point. Additional tests isolate the fault to within 5 chips in 1 case, and 2 chips in 2 cases for this condition. For the condition where all data is zero at test point D32, then the fault is in 1 of the 10 remaining chips and no farther fault isolation can be performed at test point D32. Unfortunately, none of the remaining internal test points that are available at the test bus can significantly improve the degree of fault isolation. Instead ELCK which is the dot clock signal at the output of U83A in Fig. 2.4 is proposed as an additional test point. A simple go, no-go test of the clock signal at this point isolates the fault to 1 of 6 chips.

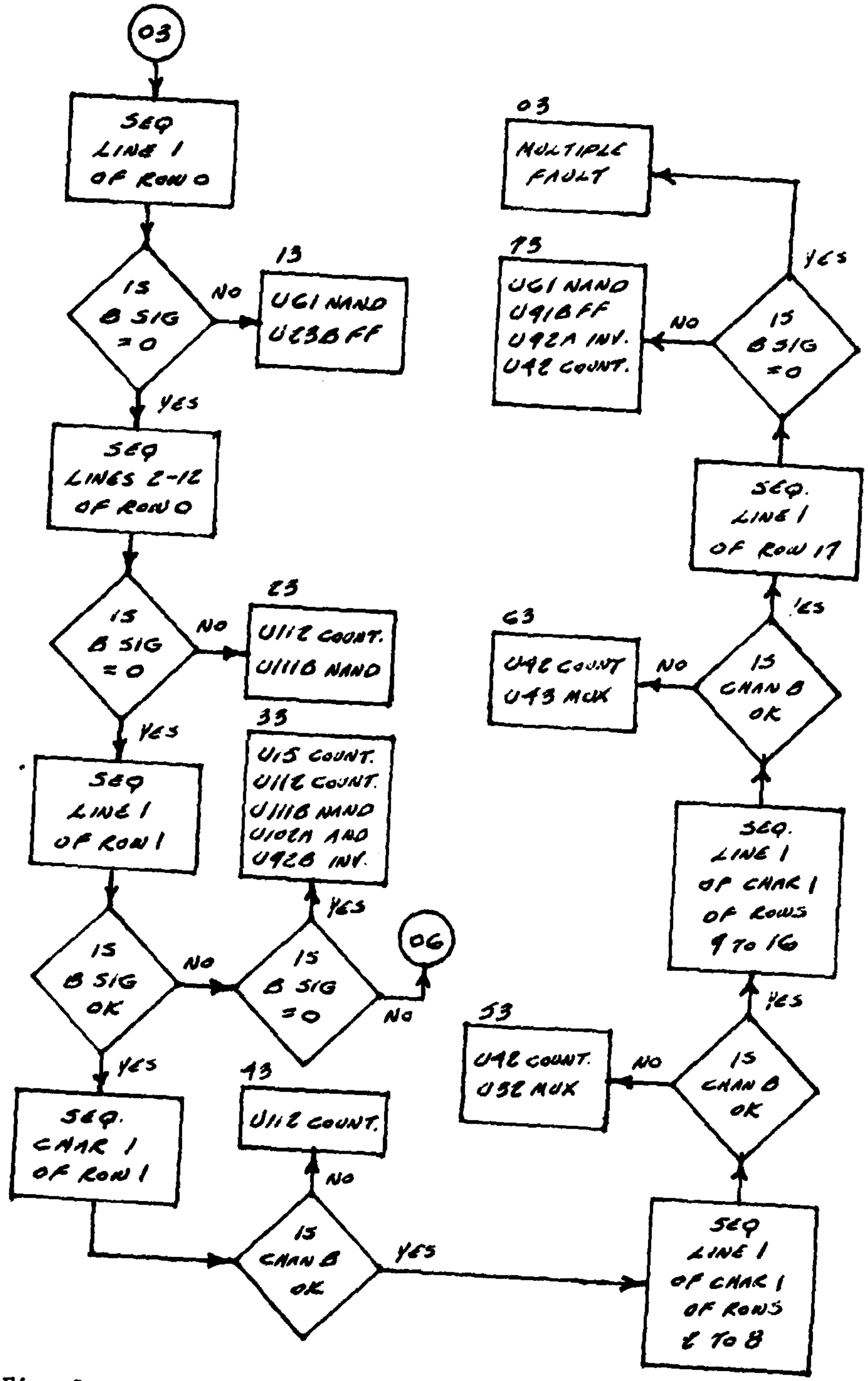

Fig. 3.10. Timing Circuit Fault Isolation Subroutine (1)

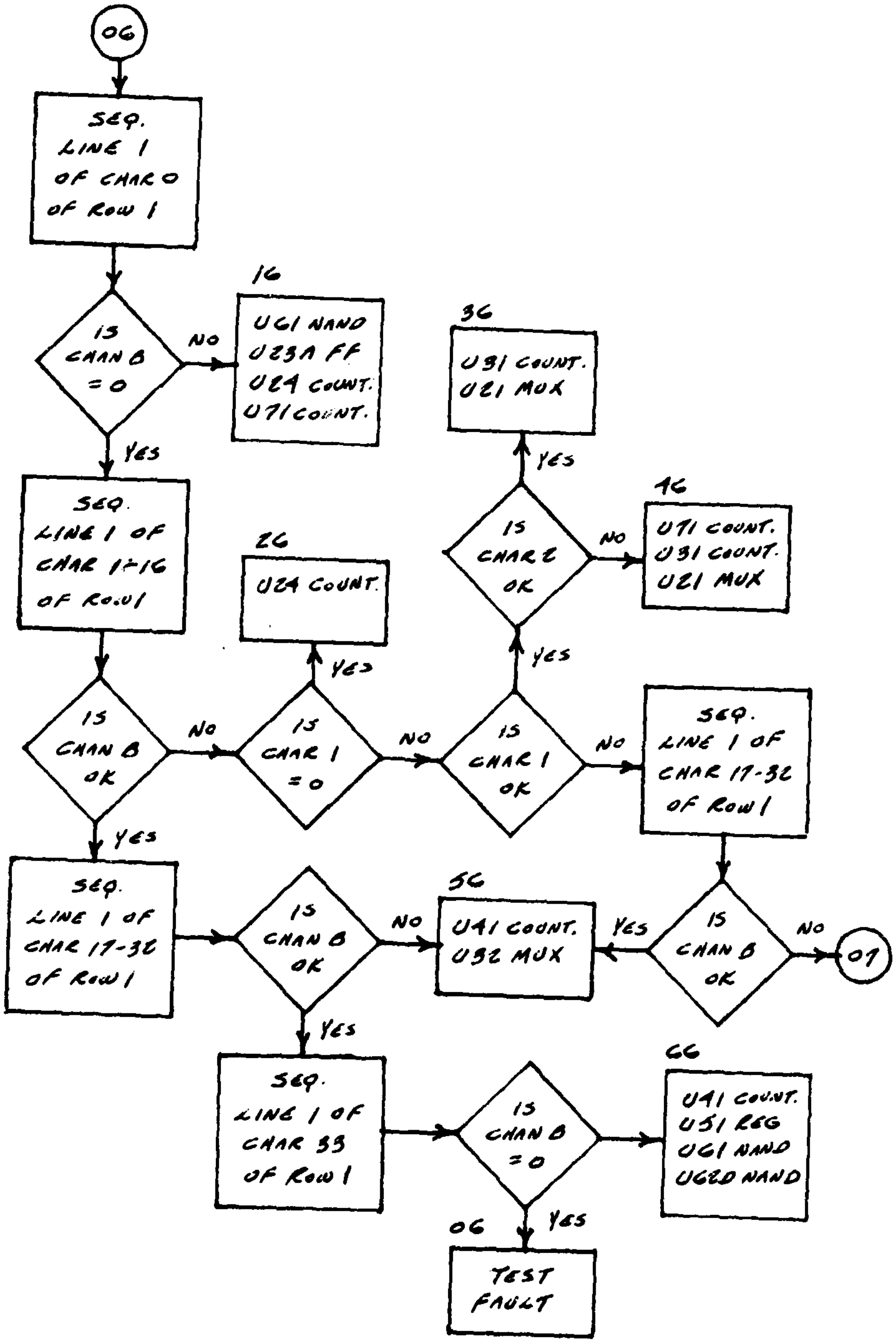

Fig. 3.11. Timing Circuit Fault Isolation Subroutine (2)

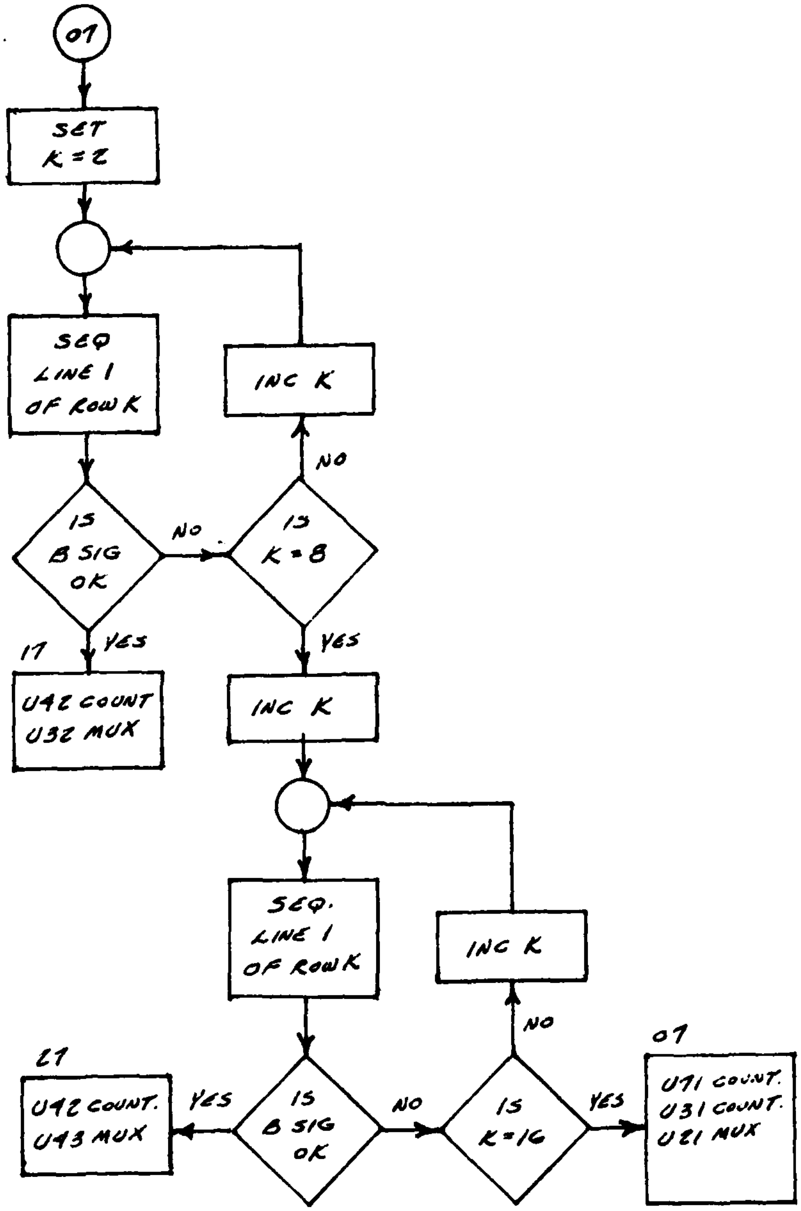

Fig. 3.12. Timing Circuit Fault Isolation Subroutine (3)

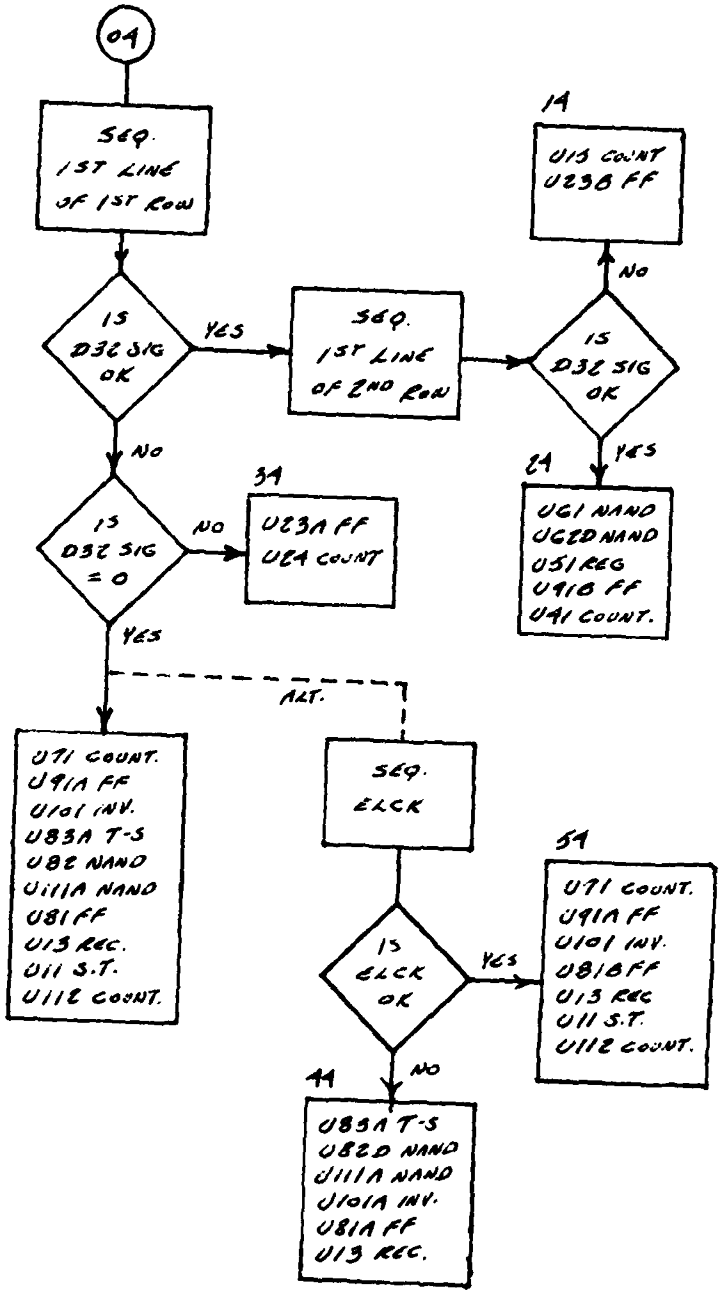

Fig. 3.13. Inhibited Data Fault Isolation Subroutine

4.0. FAULT ISOLATION TEST IMPROVEMENTS

A summary of the degree of fault isolation obtained with the test procedures of Section 3.0 is shown in Table 4-1. This lists the number of possible faulty chips for each single chip error code for both the RAM W/R and character generator tests. For the RAM W/R test, assuming that each of 44 error codes are equally probable, fault isolation is to within an average of 1.75 chips. For the character generator test, assuming that each of 37 error codes are equally probable, isolation is to within an average of 2.60 chips. The poorer isolation obtained for the character generator is partially due to the fact that only 3 internal test points are used in this test, whereas 7 are used in the RAM W/R test.

A question that naturally arises is this. Is the degree of fault isolation good enough and what can be done to improve it? One method of improving fault isolation is to add internal observation test points and diagnostic tests. This may or may not be cost/time-effective depending on the application. Instead we will consider two designs for test rules that improve isolation without increasing the number of internal test points or diagnostic time. Specifically, if these were implemented, the number of possible faulty chips for 8 error codes would be reduced to the figures shown in parenthesis in Table 4-1. Average fault isolation would be reduced to 1.59 chips for the RAM W/R test and 2.51 chips for the character generator test.

4.1 Design for Test Rules

The first and most important design for test rule is to minimize the number of components used in the system. This is an old fashioned design rule which has largely been neglected since the widespread use of MS1 and LS1. While the use of MS1 and LS1 components certainly tends to minimize chips, standard minimization techniques can still be used to reduce the number of SS1 components required.

As an example of how minimization can improve fault isolation, consider the mod-12 line counter shown in Fig. 4.1a. This uses all or parts of 4 separate chips, U112, U111B, U102A, and U92B. Consequently, isolating certain faults to the line counter means that the fault can exist in any of these 4 chips. However, making use of don't cares, inverter U92B can be eliminated and U111B becomes a 3 input rather than 4 input NAND gate. Also,

TABLE 4-1. FAULT ISOLATION TEST SUMMARY

RAM W/R Test		Char. Ges. Test.	
Error Code	# of Chips	Error Code	# of Chips
05	2	11, 12	2
0A	2	21, 22	3
11, 12, 14, 18	1	31,32	2
21, 22, 24, 28	1	41, 42	3
31, 32, 34, 38	1	51, 52	2
1F	1	61, 62	1
2F	4	71, 72	2
3F	2	81, 82	2
4F	3-(2)	13	2
5F	3-(2)	23	2
6F	1	33	5-(3)
7F	2	43	1
8F	2	53	2
9F	1	63	2
0F	4	73	4
03	2	16	4
13	3-(2)	26	1
23	1	36	2
33	1	46	3
43	2	56	2
53	1	66	4
63	1	07	3
0C	6-(5)	17	2
1C	1	27	2
2C	2	14	2
10	1	24	5
20	2	34	2
30	2-(1)	44	6-(5)
40	2-(1)	54	6
50	1		
60	1		
70	2-(1)		
80	3		
90	2		
A0	2		
Ave.	1.75	Ave.	2.60

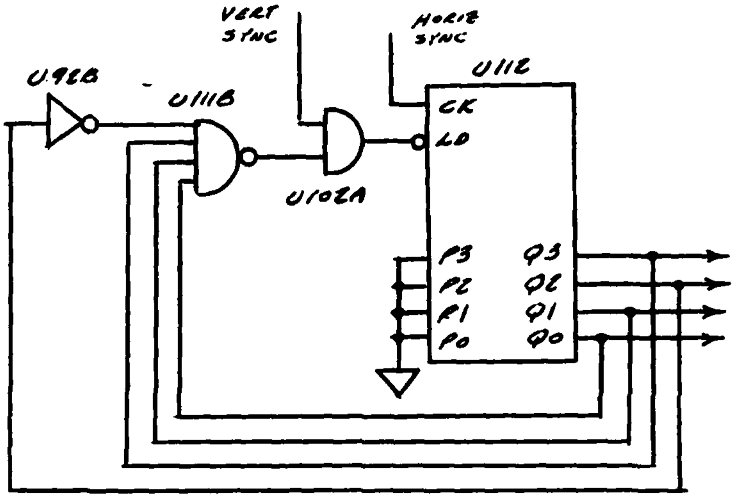

Fig. 4.1a. Line Counter

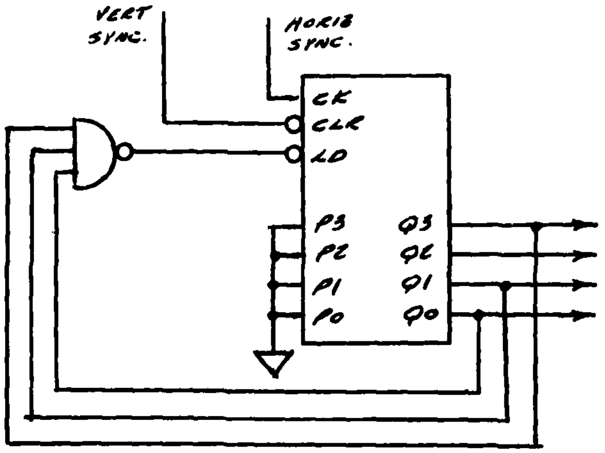

Fig. 4.1b. Simplified Line Counter

making use of the synchronous clear input of a 74163 programmable counter (instead of a 74161) eliminates the need for the U102A AND gate. The simplified mod-12 counter, which uses only parts of 2 chips, is shown in Fig. 4.1b. This circuit, which is functionally equivalent to the original circuit, improves line counter fault isolation by a factor of 2 in some cases.

Next consider the RAM read/write/decode/MUX logic shown in Fig. 2.2. The 8131 bus comparator U12 in cascade with inverter U101E is equivalent to an 8136 bus comparator. Thus inverter U101E can be eliminated. Inverter 92C can also be eliminated, since it performs the same function as inverter U92F, and there are no fan-out problems. Similarly, AND gate U102B can be eliminated since it performs the same function as U102C. Two AND gates which have been eliminated, U102A and U102B, can be used to replace NAND gate U82B in cascade with inverter U92E and NAND gate U82C in cascade with inverter U92D. The net result of these changes is the elimination of 5 inverters and two 2 input NAND gates. Since one hex inverter chip uses only 5 inverters, this chip can be eliminated. Additional chips could be eliminated if certain gates that are used for test purposes only were eliminated. However, these will not be considered here. The important thing is the improvement in the degree of fault isolation that can be obtained without interfering with any system or manufacturer's test function.

The second design for test rule is this. Whenever, two similar gates are connected in cascade, they should be located on the same chip. If this is the case they can be considered as a single block for fault diagnostic purposes, rather than as two separate blocks. The only example that violates this simple rule in the text generator is the cascade of the 2 input NAND gates U82D and U111A in Fig. 2.4.

The intent of presenting these design change examples is not that they be made for the text generator. Once the system has been designed, it is too late to make expensive design changes. Rather these are intended to illustrate how these rules can improve fault isolation if they are implemented during the initial design process.

5.0. CONCLUSIONS

This study has resulted in the formulation of a unique test philosophy along with a complete set of fault diagnosis flow diagram for the text generator. These were designed to minimize diagnostic time and the number of interfaces between the system under test and the test processor. Although this information is useful in itself, this study was originally intended as a first step in obtaining a working breadboard of a microprocessor-based automatic fault diagnosis system. The Electrical Engineering Department of the University of South Florida has purchased a Synertek SYM-I single board computer for this purpose. However, because of time limitations during the present study, much remains to be done to accomplish this goal. Specific tasks are as follows:

1) Define and purchase circuits to interface the test MPU with the text generator as required.
2) Translate fault diagnosis flow diagrams into MPU machine language programs and debug as necessary.
3) Add programs for self-test of test MPU and interface circuits.

Other extensions of this research study that could be more useful than completing a working breadboard automatic fault diagnosis system for the text generator are as follows.

1) At the present time, the fault diagnosis flow diagrams are designed to isolate all single chip faults but only a limited number of double chip faults. It would be extremely useful if these were extended to cover all double chip faults.
2) It would also be extremely useful to apply the test philosophy and formulate a complete set of fault diagnosis flow diagrams for a more complex digital subsystem, e.g., a small computer.

6.0. REFERENCES

[1] Frokwerk, Robert A., "Signature Analysis: A New Digital Field Service Method," Hewlett-Packard Journal, May 1977.

[2] Breuer, M. A. and Friedman, A. D., "Diagnosis and Reliable Design of Digital Systems," Computer Science Press, Inc., 1976.

DATE
ILMED

— 8